AF INSTITUTE OF TECHNOLOGY

# Frequency Assignments for HFDF Receivers in a Search and Rescue Network

THESIS

Krista E. Johnson
Captain

AFIT/GOR/ENS/90M-9



DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GOR/ENS/90M-9

# Frequency Assignments for HFDF Receivers in a Search and Rescue Network

THESIS


Krista E. Johnson
Captain


AFIT/GOR/ENS/90M-9

AFIT/GOR/ENS/90M-9

# Frequency Assignments for HFDF Receivers in a Search and Rescue Network

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science (Operational Sciences)

Krista E. Johnson, B.S.
Captain



March, 1990

## *Preface*

The following document represents the culmination of a six month research effort that was aided immensely by the time, patience, and assistance of many people.

First, I would like to extend my gratitude to my thesis advisors, Dr. James W. Chrissis and Dr. Yupo Chan. Their guidance and perspective is deeply appreciated. I would also like to thank my thesis sponsors, Dr. Alfred Marsh and Capt David Drake, for presenting me with such a challenging and interesting problem, as well as for their assistance during the research effort. A special thanks is due to all of the computer systems operators at AFIT, especially Mr. Doug Burkholder, Sgt Vivian Crisp, Ms. Kristin Larsen, and Mr. Jack Phillips. Their assistance was invaluable.

Finally, I would like to thank all my friends and family (you know who you are) for all of their patience and support. I couldn't have done it without them.

Krista E. Johnson

*Table of Contents*

*List of Figures*

AFIT/GOR/ENS/90M-9

## *Abstract*

This thesis applies a multiobjective linear programming approach to the problem of assigning frequencies to HFDF receivers in a search-and-rescue network in order to maximize the expected number of geolocations of vessels in distress. The problem is formulated as a multiobjective integer linear programming problem. The integrality of the solutions is guaranteed by the totally unimodularity of the $A$-matrix.

Two approaches are taken to solve the multiobjective linear programming problem: 1) The first is the multiobjective simplex method as implemented in ADBASE; and 2) a FORTRAN program developed by Dr. Ralph Steuer, University of Georgia. The second is an iterative approach. In this approach, the individual objective functions are weighted and combined in a single additive objective function. The resulting single objective problem is expressed as a network programming problem and solved using SAS NETFLOW. The process is then repeated with different weightings for the objective functions. The solutions obtained from the multiobjective linear programs are evaluated using a FORTRAN program provided by the Department of Defense to determine which solution provides the greatest expected number of geolocations. This solution is then compared to the sample mean and standard deviation for the expected number of geolocations resulting from 10,000 random frequency assignments for the network.

The best solution obtained using the multiobjective integer linear programming approach provided an approximate expected number of geolocations that was more than 13 standard deviations better than the mean approximate expected number of geolocations obtained from 10,000 randomly frequency assignments. Thus, the solutions obtained using this approach are significantly better than could be expected from an arbitrary frequency assignment. Theses, (EDC)

# Frequency Assignments for HFDF Receivers in a Search and Rescue Network

## I. Introduction

### 1.1 Background

The United States has a worldwide network of listening stations with a search and rescue mission covering large ocean areas. The purpose of this network is to locate ships and planes in distress so that search and rescue (SAR) operations can be inititated (11:1).

Each of the stations has equipment to receive and process distress signals and to compute a line of bearing (LOB) to the plane or ship transmitting the signal. While two LOBs can be used to estimate the position of the vessel in distress, grossly inaccurate position estimates can be calculated if one of the bearings is wrong. Therefore, at least three LOBs from different stations are required before the location of the vessel can be estimated with an acceptable level of confidence (11:1).

There are two types of location estimates. The first, called a "point estimate," is an estimate of the exact location of the vessel. The second, called a "confidence region," is the elliptical region that probably contains the vessel in distress. If the confidence region is too large, SAR operations will not be conducted. The size of the confidence region depends on the geography of the stations providing the LOBs and the smallest acceptable probability of the confidence region actually containing the vessel (11:2,7).

Two types of receiving systems are used in the location process, the Receiving Subsystem (RS) and the High Frequency Direction Finding (HFDF) subsystem. A

signal that has been detected and processed by an RS is said to be "acquired," while a signal that has been detected and processed by an HFDF is said to be "received" (11:4).

The location process begins when a signal has been acquired. An alarm sounds and an RS operator responds by notifying Central Control. Central Control then sends a message to the other stations asking them to check their RS and HFDF subsystems for the signal in question and to transmit a LOB to Central Control if one is available. If at least three LOBs are available, Central Control calculates a point estimate and a confidence region for the vessel transmitting the distress signal. If the confidence region is sufficiently small, SAR operations are initiated (11:2).

One RS and anywhere from zero to ten HFDFs are located at each station. The RS monitors the entire range of frequencies. When a signal is detected by the RS, an alarm is sounded and information about the signal is stored. Each HFDF, on the other hand, can monitor only one preset frequency band. An HFDF is more sensitive than an RS, however, and can detect a weaker signal. When a signal is detected by an HFDF, information about the signal is stored, but no alarm is sounded (11:1-2,4).

The probability of successfully locating a vessel in distress depends in part on the frequencies that are assigned to the various HFDF receivers. However, determining the optimal assignment of frequencies to HFDF receivers is an extremely difficult task (11:2-3). For example, suppose there are 30 frequencies that may be assigned to an HFDF. If there are six HFDFs located at a station, there are more than 500,000 possible frequency assignments for that station. If there are 30 frequencies and three receiving stations, each with six HFDFs, there are more than $10^{17}$ possible frequency assignments. Obviously, complete enumeration and comparison of all the possible frequency assignments is not practical even with the most powerful computers in existence. Therefore, alternative methods of HFDF receiver assignment must be explored (11:3).

### 1.2 Notation

The following notation is used throughout this thesis.

Subscripts: Let $i$ refer to transmitting locations,
$j$ refer to receiving stations,
$k$ refer to frequencies.

Decision Variables: Let $x_{jk} = \begin{cases} 1 & \text{if an HFDF receiver at station } j \\ & \text{is assigned to frequency } k \\ 0 & \text{otherwise} \end{cases}$

Probabilities: Let $F_{ik}$ be the probability of a distress signal from transmitting location $i$ being made on frequency $k$.

$PD_{ijk}^{RS}$ be the probability that a signal transmitted from location $i$ on frequency $k$ propagates to station $j$ with sufficient signal-to-noise ratio for detection by RS.

$PD_{ijk}^{DF}$ be the probability that a signal transmitted from location $i$ on frequency $k$ propagates to station $j$ with sufficient signal-to-noise ratio for detection by HFDF.

$PP_{jk}^{RS}$ be the probability that a signal detected by RS on frequency $k$ at station $j$ is processed on RS.

$PP_{jk}^{DF}$ be the probability that a signal detected by HFDF on frequency $k$ at station $j$ is processed on HFDF.

Other Notation: Let $m_j$ be the number of HFDF receivers at receiving station $j$

$d_i$ be the largest acceptable confidence region radius for transmitting location $i$

$\alpha$ be any combination of three or more receiving stations

$C$ be the set of all combinations of three or more receiving stations

$$I_{i\alpha} = \begin{cases} 1 & \text{if combination } \alpha \text{ yields a confidence region radius} \leq d_i \\ & \text{for transmitting location } i \\ 0 & \text{otherwise} \end{cases}$$

### *1.3 Probabilities of Acquiring and Receiving Distress Signals*

The following probabilities are used in the formulations of the problem. A signal is said to be "acquired" when it has been detected and processed by an RS. The probability, $P_{ijk}^{RS}$ that a signal transmitted on frequency $k$ from location $i$ is acquired at receiving station $j$ is given by the product of the probability that the signal propagates to station $j$ with sufficient signal-to-noise ratio for detection by the RS and the probability that a signal that has been detected at station $j$ on RS is processed on RS. In mathematical terms,

$$P_{ijk}^{RS} = PD_{ijk}^{RS} \cdot PP_{jk}^{RS}$$

A signal is said to be "received" when it has been detected and processed by an HFDF. Note that a signal can only be received if an HFDF receiver is assigned to the signal's frequency. The probability, $P_{ijk}^{DF}$ that a signal transmitted on frequency $k$ from location $i$ is received at receiving station $j$ is given by the product of the probability that the signal propagates to station $j$ with sufficient signal-to-noise ratio for detection by HFDF, the probability that a signal that has been detected at station $j$ on HFDF is processed on HFDF, and $x_{jk}$. In mathematical terms,

$$P_{ijk}^{DF} = PD_{ijk}^{DF} \cdot PP_{jk}^{DF} \cdot x_{jk}$$

Since a higher signal-to-noise ratio is required for a signal to be detected by RS than is required for detection by HFDF, a signal detected by RS will also be detected by HFDF at that station if an HFDF receiver is assigned to the signal's transmitting frequency. The probability that a signal is detected and processed by both RS and HFDF is thus given by the probability that the signal propagates to station $j$ with enough signal to noise ratio that it is detected by RS, the probability that a signal that has been detected at station $j$ on RS is processed on RS, the probability that a signal that has been detected at station $j$ on HFDF is processed on HFDF, and $x_{jk}$. In mathematical terms,

$$PD_{ijk}^{RS} \cdot PP_{jk}^{RS} \cdot PP_{jk}^{DF} \cdot x_{jk}$$

The probability that a signal is detected and processed by either RS or HFDF, $S_{ijk}^{RS/DF}$, is given by the probability that the signal is acquired plus the probability that the signal is received minus the probabililty that it is both acquired and received. In mathematical terms,

$$S_{ijk}^{RS/DF} = PD_{ijk}^{RS} \cdot PP_{jk}^{RS} + PD_{ijk}^{DF} \cdot PP_{jk}^{DF} \cdot x_{jk} - PD_{ijk}^{RS} \cdot PP_{jk}^{RS} \cdot PP_{jk}^{DF} \cdot x_{jk}$$

The probability that a signal is detected and processed by HFDF only, $S_{ijk}^{DF}$, is given by the probability that the signal is received minus the probability that it is both acquired and received. In mathematical terms,

$$S_{ijk}^{DF} = PD_{ijk}^{DF} \cdot PP_{jk}^{DF} \cdot x_{jk} - PD_{ijk}^{RS} \cdot PP_{jk}^{RS} \cdot PP_{jk}^{DF} \cdot x_{jk}$$

### *1.4 Mathematical Formulation*

This section presents the mathematical formulation of the frequency assignment problem for HFDF receivers in the SAR network. This formulation, suggested by Drake (11:5-12), is a faithful model of the complete problem and can be stated as:

Maximize the expected number of geolocations by the SAR network subject to limited HFDF resources. A geolocation requires that a signal be detected and processed by at least one RS and by at least three receiving stations and that the confidence region radius be smaller than $d_i$.

*1.4.1 Objective Function.* The probability of a transmission being made from transmitting location $i$ on frequency $k$ can be thought of as the expected number of signals transmitted. Therefore, the expected number of geolocations is given by

$$\sum_i \sum_k F_{ik} \times \left[ \begin{array}{l} \text{Probability of detecting and processing the signal by at least one RS} \\ \text{and by at least three receiving stations} \\ \text{and that the confidence region radius is smaller than } d_i \end{array} \right]$$

The probability of detecting and processing a signal by a particular combination of receiving stations and by at least one RS in that combination is obtained by subtracting the probability that all the stations in the combination process and detect the signal on HFDF only from the probability that all stations in the combination detect and process the signal on either RS or HFDF and multiplying that quantity by the probability that all other stations do not detect and process the signal. In mathematical terms,

$$\left[ \prod_{j \in \alpha} S_{ijk}^{RS/DF} - \prod_{j \in \alpha} S_{ijk}^{DF} \right] \left[ \prod_{h \in \alpha'} (1 - S_{ihk}^{RS/DF}) \right]$$

Also, for any particular combination of three or more stations, the function $I_{i\alpha}$ takes on the value one if the confidence region radius is less than $d_i$, and zero otherwise. Thus, the requirement that the radius of the confidence region be less than $d_i$ is satisfied by multiplying the quantity given above by $I_{i\alpha}$.

Since at least three stations must detect and receive the signal, these probabilities are summed over all combinations of three or more stations.

This yields the complete objective function:

$$\text{Maximize } \sum_i \sum_k F_{ik} \sum_\alpha \left[ \prod_{j\in\alpha} S_{ijk}^{RS/DF} - \prod_{j\in\alpha} S_{ijk}^{DF} \right] \left[ \prod_{h\in\alpha'} (1 - S_{ihk}^{RS/DF}) \, [I_{i\alpha}] \right]$$

*1.4.2 Constraints.* There are only two constraints in this formulation. The first constraint is that the number of HFDF frequency assignments made to any particular receiving station is the same as the number of HFDF receivers located at that station. In mathematical terms,

$$\sum_k x_{jk} = m_j \text{ for all } j$$

The second constraint is that only one frequency can be assigned to each HFDF receiver. In mathematical terms,

$$x_{jk} = 0 \text{ or } 1$$

*1.4.3 Tractability.* Although this formulation of the problem encompasses the entire problem, it is also computationally intractable. This is due to the complexity of the objective function and the requirement that the decision variables be equal to zero or one.

### *1.5 Research Objective*

The objective of this research is to determine the optimal, or nearly optimal assignment of frequencies to the HFDF receivers for the SAR problem using a mathematical programming approach.

#### *1.5.1 Subobjectives.*

1. Select a mathematical programming model to describe the problem and identify appropriate solution techniques.

2. Formulate the problem in terms of the mathematical programming model selected.
3. Assign frequencies to the HFDF receivers.
4. Evaluate the frequency assignments.
5. Validate the use of the model.

### *1.6 Overview*

The thesis effort is described in detail in the remaining chapters. A description of the problem formulation, as well as an overview of the literature that contributed to and motivated the formulation is provided in Chapter 2. The methodologies employed in the solution of the problem are described in Chapter 3 and the actual results are discussed in Chapter 4. Finally, validation of the model and conclusions are presented in Chapter 5.

# *II. Problem Formulation*

This chapter presents the simplified formulation of the problem, as well as an overview of the literature that contributed to the formulation. The first section describes the basic concepts of network programming. Location models and their applicability to the assignment of HFDF receivers in the SAR network are discussed in the following section. This is followed by an overview of several techniques used to deal with uncertainty in location models. The chapter concludes with a description of the problem formulation.

## *2.1 Networks.*

The assignment of frequencies to HFDF receivers in the SAR network may be modelled and solved using techniques generally used in network and location modelling. This section introduces the basic concepts of network models.

Network programming problems are a category of linear programming problems that have a special structure (2:353) that allows them to be represented as a network. Like all linear programming problems, they consist of a linear objective function and linear constraints. Many efficient solution techniques for combinatorial optimization problems have been developed that take advantage of this network structure.

A network is a set of nodes (points) connected by a set of arcs with some kind of flow on the arcs. A simple example of a network is a road system; the intersections are the nodes, the roads are the arcs, and the cars are the flow (12:297). If an objective is introduced, such as maximizing the amount of flow on the network or minimizing the cost of sending a given amount of flow through the network, the result is a network programming problem. The problem is generally constrained by the capacity of the arcs and by conservation of flow requirements (flow into a node must equal flow out of that node).

There are many well-known types of network problems including scheduling problems, assignment problems, routing problems, and location problems. The assignment of frequencies to HFDF receivers is, strictly speaking, an assignment problem. However, as shall be seen in the following section, it can also be thought of as a location problem.

## *2.2 Location Models.*

Location problems consist of problems where a set of service facilities are located to satisfy the demands of a geographically distributed set of customers or a set of customers are allocated to a preexisting set of service facilities. The objective is usually to minimize a function of the distances or travel times between the service facilities and the customers. The specific objective may be to minimize the average distance or travel time between the facilities and the customers, to minimize the maximum distance any customer must travel to a service facility, or to minimize some cost function of distance or travel time (4:646-647).

Some variations of the location problem include construction costs associated with building facilities at the various locations, while others involve service facilities that can only satisfy a limited amount of demand. Still others allow some of the problem parameters to be probabilistic.

Location problems are typically represented by a network where the service facilities and the customers are represented by nodes and the possible connections between them are modelled as arcs. A cost corresponding to some function of distance or travel time is usually associated with each arc.

*2.2.1 Location Models and the Search and Rescue Problem.* The assignment of frequencies to HFDF receivers for the SAR problem may be modelled as a location problem. In this case, the receiving stations can be modelled as service facilities and the various combinations of transmitting locations and frequencies can be modelled

as customers. The probabilities of a transmission being made from various locations on various frequencies may be thought of as demand. The problem of satisfying demand is complicated, however, by the requirement that a signal be detected and processed by at least one RS receiver and at least three receiving stations before the location of the vessel in distress is estimated and by the requirement that the confidence region be small. An additional complication arises since whether or not a signal is acquired by RS or received by HFDF at the various stations is also described by probabilities and these probabilities are only partially dependent on distance. A number of researchers have considered probabilistic variations of the location problem over the past several years. Some of these research efforts are discussed in the following sections.

*2.2.2 Probabilistic Location Models.* A large body of literature is available pertaining to location models. However, only a small portion of that literature pertains to probabilistic models (4, 7:645-674,95-98). A few probabilistic location models and their potential applicability to the SAR problem are discussed in this section.

*2.2.2.1 Probabilistic Demand (Model 1).* Louveaux and Thisse (16:145-149) consider a location model with probabilistic demands. The problem involves locating a production facility when market conditions (demands) are uncertain. Louveaux and Thisse formulate the problem as a two-stage stochastic model. The first stage of the model consists of deciding on the location of the facilities and their production levels before actual demands are known. The second stage involves determining the distribution of the goods produced based on demonstrated demand after the facilities have been built.

Louveaux and Thisse were interested in determining if the Hakimi theorem would hold under probabilistic market conditions. The Hakimi theorem states that "given a finite set of transportation cost functions concave in distance, the set of

nodes and market-places contains a location minimizing the total transportation cost" (16:145). The important implication contained in the theorem is that intermediate locations on the network do not have to be considered when searching for the optimal location for a facility and that the number of locations that must be considered is finite (16:145). Louveaux and Thisse found that the theorem holds when a company is willing to accept risk. However, when the company is risk averse, the theorem no longer holds and the optimal location may be on one of the arcs (16:146-149).

Although demand is uncertain in the SAR problem, the receivers can only be located at the receiving stations. Therefore this area of research appears to have minimal applicability to the problem.

*2.2.2.2 Probabilistic Demand (Model 2).* Bertsimas considers a travelling salesman facility location problem with probabilistic demands (3:146-160). In this problem, a salesman receives calls for service at the begining of each day. The objective is to determine the best routing strategy, as well as the best home location for the salesman, in order to minimize the total expected travel distance. Bertsimas notes that the optimal home location for the salesman will be on one of the nodes. The emphasis, however, is on finding the best routing strategy rather than finding the best location.

***2.2.2.3 Probabilisitic Cost and Demand.*** Seppala (19:56-62) considers a stochastic multi-facility location problem where the costs and the demands are probabilistic. In this problem, one or more new facilities are to be located among several existing facilities. The objective is to minimize the value, $t$, where the probability of the total costs being less than $t$ is greater than some set value. Seppala formulates the problem with a nonlinear objective function and nonlinear constraints. He uses a "Chance-Constrained Programming System (CHAPS)" (19:59) to solve the problem. CHAPS is an iterative process that converts the problem to a linear problem

and solves the resulting problem. Each iteration involves discarding inactive constraints and more accurately estimating the remaining constraints with new linear constraints. The process is terminated when two successive iterations are almost equal.

*2.2.2.4 Probabilistic Response to Demand (Model 1).* Daskin (6:48-70) developed a variant of the maximal coverage location problem that accounts for the possibility that a facility covering a specific node may not be able to respond to demands at that node. For example, an ambulance that is assigned to cover a particular area may not be able to respond to a call from that area because it is already responding to another call or because it has "broken down" (6:50,52). This is analogous to the situation in the SAR problem where whether or not a signal is detected by RS or HFDF at the various stations is described by probabilities. Daskin assumes the probability, $p$, that a service facility will not be able to respond to a demand is known and that it is the same for all facilities. This assumption allows the problem to be formulated with a linear objective function (6:52-53).

Daskin also developed a heuristic solution algorithm to solve this problem (6:58-60). The basic principle is to begin with the facilities located at some set of possible locations referred to as the "current solution" and then to relocate the facilities, one at a time, to obtain a "trial solution." At any time, the current solution and the trial solution will differ only by the location of one facility. Throughout the algorithm, $p^*$ is used to indicate the upper bound and $p^{**}$ to indicate the lower bound of the interval for which the current solution is the best solution identified.

The algorithm is started by setting $p^*$ equal to one, setting $p^{**}$ equal to zero, and locating all of the facilities at the node that covers the most demand. Each subsequent step consists of relocating one facility to obtain a trial solution and evaluating that solution. The following three cases may result from the evaluation.

Case 1: The trial solution is worse than the current solution for all values of $p$ between $p^*$ and $p^{**}$. In this case the trial solution is discarded.

Case 2: The trial solution is better than the current solution for $p$ equal to $p^*$. In this case, the trial solution replaces the current solution and $p^{**}$ is reset to zero.

Case 3: The trial solution is better than the current solution for some value of $p$ between $p^*$ and $p^{**}$. In this case, the trial solution becomes the "tentative solution" and $p^{**}$ is set equal to the value of $p$ below which the trial solution becomes better than the current solution.

After all trial solutions that can possibly be generated from the current solution have been evaluated, the current solution is replaced by the tentative solution if one exists, $p^*$ is set equal to $p^{**}$, $p^{**}$ is set to zero, and the procedure is repeated. If there is no tentative solution, the algorithm is terminated. Although Daskin presents evidence that the algorithm provides good solutions, he notes that an optimal solution is not guaranteed (6:58).

*2.2.2.5 Probabilistic Response to Demand (Model 2).* Hogan and Revelle (13:1434-1444) suggest a multiobjective approach to the maximal coverage location problem in the context of emergency vehicle location. The goal is to maximize the probability of a server being available to respond to a given demand while minimizing the total number of servers required. In order to maximize the probability of a server being available, Hogan and Revelle advocate backup coverage, or coverage of a demand more than once. This approach recognizes the fact that the primary server assigned to respond to a demand may not be available. Obviously, if backup coverage is provided, the probability of a server being available increases. The overall result in this approach is that backup coverage is maximized (perhaps weighted by the population of the demand node(s) receiving backup coverage) rather than attempting to directly maximize the probability of a server being available to respond to a demand. This also results in a linear objective function rather than a nonlinear

function.

### *2.3 Potential Carryovers to the Search and Rescue Problem.*

The assignment of frequencies to HFDF receivers for the SAR problem may be modelled as a location problem on a network. However, the probabilistic nature of the problem significantly complicates the formulation and solution. A number of researchers have considered probabilistic variations of the location problem over the past several years. Of the approaches suggested by the literature, Hogan and Revelle's multiobjective backup coverage model appears to show the greatest potential.

### *2.4 Simplified Problem Formulation.*

While the complete formulation of the problem presented in Chapter 1 fully describes the problem, it is computationally intractable. In this formulation, the goal is to simplify the problem sufficiently so that it can be solved while maintaining enough of the essence of the problem to yield nearly optimal results. A variation of a multiobjective covering model was identified as a promising modelling approach during the literature search. Thus, the problem is formulated as a zero-one multiobjective linear program with covering constraints. The research goal was to use this modelling technique to develop a heuristic to generate good HFDF frequency assignments.

*2.4.1 Objective Function.* A multiobjective linear programming approach was chosen to model the problem. In other words, more than one linear objective function is used to generate an HFDF frequency assignment. Each one of these objective functions contribute to the solution. However, because of the complexity of the problem, a numerical relationship between the individual contributions of these objectives is not clear, and indeed may be different for different transmission and reception probabilities.

*2.4.1.1 Objective Function 1.* The first objective is to maximize the expected number of lines of bearing provided by HFDF only. In other words, the objective is to maximize multiple coverage of a frequency weighted by the probability that transmissions made on that frequency will be detected and processed by HFDF only at any particular receiving station. This is similar to Hogan and Revelle's backup coverage approach weighted by the probability of a single server being able to respond to a demand.

This objective function is motivated by the fact that there is little benefit to be gained by assigning an HFDF receiver at a station to receive and detect a signal if there is a large probability that the signal will be acquired by RS at that station. On the other hand, if there is a large probability of a signal being received by HFDF only at a particular station, there is a large probability of obtaining a line of bearing from HFDF for that signal. Maximizing the number of lines of bearing provided for the various transmissions should tend to maximize the number of signals that are actually detected and processed by three or more receiving stations.

Also, the size of the confidence region tends to decrease as the number of stations acquiring or receiving the signal increases. Thus, maximizing the expected number of lines of bearing provided for the various transmissions should tend to reduce the size of the resulting confidence region. Therefore, the requirement that a confidence region be less than $d_i$ is not explicitly addressed in this formulation.

The development of the first objective function in mathematical terms is:

$F_{ik}S_{ijk}^{DF} =$ E[number of transmissions from transmitting location i on frequency k received by HFDF only at station j]

$\sum_j F_{ik}S_{ijk}^{DF} =$ E[number of LOBs provided by HFDF only for transmissions from transmitting location i on frequency k]

$$\sum_j \sum_k F_{ik} S_{ijk}^{DF} = \begin{array}{l}\text{E[number of LOBs provided by HFDF only for} \\ \text{transmissions from transmitting location i]}\end{array}$$

$$\sum_i \sum_j \sum_k F_{ik} S_{ijk}^{DF} = \text{E[number of lines of bearing provided by HFDF only]}$$

Thus, the first objective function is:

$$\text{Maximize} \sum_i \sum_j \sum_k F_{ik} S_{ijk}^{DF}$$

*2.4.1.2 Objective Function 2.* The second objective function is to maximize multiple coverage of the frequencies weighted by the expected use of that frequency. This objective is also similar to Hogan and Revelle's backup coverage model. However, in this case multiple coverage is weighted by the probability of a demand occurring. This objective ensures that a frequency receives its "fair share" of coverage based on the expected use of that frequency. The development of the second objective function in mathematical terms follows:

Since $\sum_j x_{jk}$ = The number of HFDF receivers assigned to frequency k, for all k, the number of HFDF receivers assigned to frequency k weighted by the expected use of frequency k can be written as

$$\sum_i \sum_j \sum_k F_{ik} x_{jk}$$

and the second objective function is thus given by

$$\text{Maximize} \sum_i \sum_j \sum_k F_{ik} x_{jk}$$

*2.4.1.3 Objective Function 3.* The third objective function is to penalize excessive coverage of the frequencies by HFDF receivers. This objective function recognizes the decreasing utility of assigning additional HFDF receivers to the various

frequencies. Without this objective function, the HFDF receivers at each receiving station would be assigned to the frequencies having the greatest probability of being received at that station or to the frequencies where the probability of a transmission is the greatest, regardless of the number of HFDF receivers already assigned to those frequencies.

This objective function is analogous to Hogan and Revelle's backup coverage model with no weighting scheme applied, although the approach used here is somewhat different. Hogan and Revelle use a separate objective function to reward each level of multiple coverage (13:1440-1444). In other words, they use one objective function to maximize second coverage of a node, another objective function to maximize third coverage of a node, and so on. The effect of using separate objective functions is to evenly distribute coverage among the nodes. The approach used in the formulation of the SAR problem, in contrast, is to use a single objective function to penalize deviation from an even distribution of coverage. The net result in either case is to maximize multiple coverage while ensuring that coverage is evenly distributed among the nodes.

This objective requires the addition of one structural variable, $y_k$, for each frequency to measure the extent of excessive coverage for that frequency. Excessive coverage is considered to be anything more than the total number of HFDF receivers divided by the total number of frequencies rounded to the next largest integer (i.e. the ceiling function, $\lceil \cdot \rceil$). In mathematical terms, excessive coverage is defined by

$$\left\lceil \frac{\sum_j m_j}{\sum_k 1} \right\rceil$$

The development of the third objective function in mathematical terms is as follows. Let

$$y_k = \begin{cases} n & \text{if frequency k is covered by } \left\lceil \frac{\sum_j m_j}{\sum_k 1} \right\rceil + n \text{ HFDF receivers} \\ 0 & \text{if frequency k is covered by } \left\lceil \frac{\sum_j m_j}{\sum_k 1} \right\rceil \text{ or less HFDF receivers} \end{cases}$$

so that

$$\sum_k y_k = \text{The amount of excess coverage in the network}$$

The complete objective function is thus:

$$\text{Minimize} \sum_k y_k$$

$$\text{or Maximize} \sum_k (-y_k)$$

*2.4.2 Constraints.* Three major sets of constraints are included in the formulation. First, the number of HFDF frequency assignments made to any particular receiving station can not exceed the number of HFDF receivers located at that station. These constraints can be expressed as:

$$\sum_k x_{jk} \leq m_j \text{ for all } j$$

Second, since none of the objective functions addresses the requirement that three LOBs are required to estimate the position of a vessel transmitting an emergency signal, it seems reasonable to require that at least two HFDF receivers be assigned to every frequency (one LOB must be provided by an RS). Mathematically, this can be expressed as:

$$\sum_j x_{jk} \geq 2 \text{ for all } k$$

Finally, excess coverage must be measured; that is, $y_k$ must equal the number of HFDF receivers in excess of $\left\lceil \frac{\sum_j m_j}{\sum_k 1} \right\rceil$ assigned to any particular frequency. This yields the following set of constraints:

$$(\sum_j x_{jk}) - y_k \le \left\lceil \frac{\sum_j m_j}{\sum_k 1} \right\rceil \text{ for all } k$$

## *2.5 Complete Simplified Problem.*

The complete problem to be solved and analyzed is:

$$\text{Maximize} \sum_i \sum_j \sum_k F_{ik} S_{ijk}^{DF}$$

$$\text{Maximize} \sum_i \sum_j \sum_k F_{ik} x_{jk}$$

$$\text{Maximize} \sum_k (-y_k)$$

subject to

$$\sum_k x_{jk} = m_j \text{ for all } j$$

$$\sum_j x_{jk} \ge 2 \text{ for all } k$$

$$(\sum_j x_{jk}) - y_k \le \left\lceil \frac{\sum_j m_j}{\sum_k 1} \right\rceil \text{ for all } k$$

$$x_{jk} = 0 \text{ or } 1 \text{ for all } j \text{ and } k$$

$$y_k \ge 0 \text{ and integer for all k}$$

The methodologies used to solve and analyze this problem are discussed in the following chapter.

# III. Solution Methodology

A variation of a multiobjective covering model was identified as a promising modelling approach during the literature search. Thus, the problem was formulated as an integer, multiobjective, linear program with covering constraints. This chapter describes the methodologies used in the solution and analysis of the search and rescue problem. Specifically, linear programming, integer programming, network programming, and multicriteria optimization are discussed.

## *3.1 Linear Programming.*

The assignment of frequencies to HFDF receivers in the SAR network was formulated as a linear programming problem. In other words all of the objective functions and all of the constraints are linear. The most popular technique for solving linear programming problems is the simplex method. The simplex method is an algebraic procedure that involves searching the boundary of the feasible region for the optimal solution to the constrained optimization problem. The feasible region is defined by the intersection of the regions defined by the constraints and corner-point solutions are the points where $n$ constraint equations intersect (where $n$ is the number of decision variables and the constraint equations include the non-negativity constraints). Two corner-point solutions are adjacent if they are connected by an edge defined by a constraint equation.

The simplex method takes advantage of three major facts (12:54-55). First, if an optimal solution exists, it is a corner-point solution. Second, there are a finite number of feasible corner-point solutions. Third, if a corner-point solution is equal to or better than all of its adjacent corner-point solutions, it is an optimal solution. The simplex method is summarized by:

1. Start at a feasible corner-point solution.

2. Move to a better adjacent corner-point solution if one is available. Repeat this step until it is no longer possible to move to a better solution.

3. If no adjacent corner-point solution is better than the current solution, stop. The current solution is optimal.

While it is theoretically possible for the simplex method to examine every feasible corner-point solution, this rarely occurs in practice. In fact, the simplex method has proven to be extremely efficient and is used in most linear programming software (17:28).

### *3.2 Integer Programming.*

The formulation of the problem restricts the decision variables, $x_{jk}$ and $y_k$, to integer values. Specifically, $x_{jk}$ must equal zero or one, while $y_k$ may take any positive integer value less than or equal to the number of receiving stations in the network.

Several techniques can be used to obtain integer solutions. One popular method for solving integer linear programming problems is to use cutting plane algorithms. This technique involves solving the linear program using the simplex method while ignoring the integer restriction. If an integer solution is obtained, the problem is solved. If not, the feasible region is reduced by adding a constraint constructed so that no feasible integer solution is discarded. The new linear problem is then solved using the dual simplex method. The process is repeated until an integer solution is identified or the problem is found to be infeasible. (2:274-275).

Another popular method is the branch and bound technique. This technique also begins by solving the linear program using the simplex method while ignoring the integer restriction. If an integer solution results, the problem is solved. If not, the objective function value for that solution provides an upper bound, $Z_U$, for the problem. The set of feasible solutions is then partitioned into subsets (branching)

and an upper bound is computed for each of the subsets. If an integer solution has been identified, its objective function value provides a lower bound, $Z_L$, for the optimal solution (bounding). Any subset whose upper bound is less than $Z_L$ is then abandoned. Subsets are also abandoned if they are found to be infeasible or if an integer solution is obtained (since no further improvement can be obtained by further partitioning the feasible region). Subsets abandoned for any of these reasons are said to be fathomed. If an integer solution is obtained whose objective function value, $Z$, is greater than the current $Z_L$, $Z$ becomes the new $Z_L$. Unfathomed subsets are further partitioned and upper bounds are computed for each of the new partitions. The process continues until all subsets are fathomed. The solution corresponding to $Z_L$ at that time, if one exists, is the optimal solution. If all subsets have been fathomed and $Z_L$ does not exist, there is no feasible integer solution (12:403-405).

*3.2.1 Total Unimodularity.* Both of the integer programming techniques discussed above begin by solving the linear problem while ignoring the integer restriction. If an integer solution is obtained, the problem is solved. Obviously, this is the most desirable result and for certain problems this result is guaranteed. Consider the integer linear programming problem given by

$$\text{Maximize } Cx$$

$$\text{subject to } Ax \leq b$$

$$x \text{ integer}$$

where $C$ is the vector of cost coefficients, $x$ is the vector of decision variables, $A$ is the matrix of coefficients for the constraints and $b$ is the vector of right hand side values for the constraints. If all of the elements of $b$ are integer and the $A$ matrix is totally unimodular, the solution will be integer (17:541). A matrix, $A$, is totally unimodular if the determinant of each square submatrix of $A$ equals zero, one, or

negative one. Fortunately, there are several theorems that can be used to determine if a matrix is totally unimodular without computing the determinant of each square submatrix of the matrix.

Any matrix, $A$, is totally unimodular if 1) all the elements of $A$ are 0, $+1$, or $-1$, 2) there are no more than two nonzero elements in any column of $A$, and 3) if there are two nonzero elements in a column, they have opposite sign (17:542). Additionally, if a given matrix is totally unimodular, other totally unimodular matrices can be constructed using the following theorem (17:540).

Let $A$ be totally unimodular, then

1. $A^T$ is totally unimodular.
2. $(A, I)$, where $I$ is an identity matrix, is totally unimodular.
3. If a row or column of $A$ is multiplied by $-1$ the resulting matrix is totally unimodular.
4. If a row or column is deleted from $A$, the resulting matrix is totally unimodular.
5. If a row or column of $A$ is duplicated, the resulting matrix is totally unimodular.
6. If two rows or columns of $A$ are interchanged, the resulting matrix is totally unimodular.
7. If pivot operations are performed on $A$, the resulting matrix is totally unimodular.

The $A$ matrix for a reduced version of the frequency assignment problem with five receiving stations and four frequencies is shown in figure 1. By applying the theorems listed above, it can be seen that the matrix is totally unimodular. Thus, integer solutions are guaranteed. Furthermore, adding the set of constraints, $x_{jk} \leq 1$, will not affect the total unimodularity of the $A$ matrix and will ensure that each $x_{jk}$ equals zero or one.

Figure 1. Totally Unimodular Matrix.

| | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | -1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | -1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | -1 |

### *3.3 Network Programming.*

Network programming is a special type of linear programming that uses efficient labelling algorithms to maximize a linear function of the flow through the network. Thus, it is desirable to express the frequency assignment problem in terms of a network. While the problem must have a special structure that allows it to be represented as a network, "nearly all" totally unimodular matrices are network matrices (17:561). The major disadvantage in using the technique for the multiobjective formulation of the search and rescue problem is that network programming is intended for single objective problems. Therefore, to use network programming techniques, some weighting scheme must be applied to the objective functions so they can be expressed as a single objective.

It is easy to visualize the assignment of frequencies to HFDF receivers as a network (see figure 2). The receiving stations and the frequencies can be thought of as nodes with the arcs between them depicting the possible assignment of HFDF receivers to the various frequencies. Flow into the network is equal to the total number of HFDF receivers, with "supply" at each receiving station equal to the number of HFDF receivers at that station. The arcs between the receiving stations and the

frequencies are capacitated with maximum flow equal to one. The cost associated with the arcs is the corresponding coefficient of the first objective function multiplied by the weight, $\lambda_1$, assigned to the first objective function plus the corresponding coefficient of the second objective function multiplied by the weight, $\lambda_2$, assigned to the second objective function. In mathematical terms, the cost associated with each of these arcs is given by:

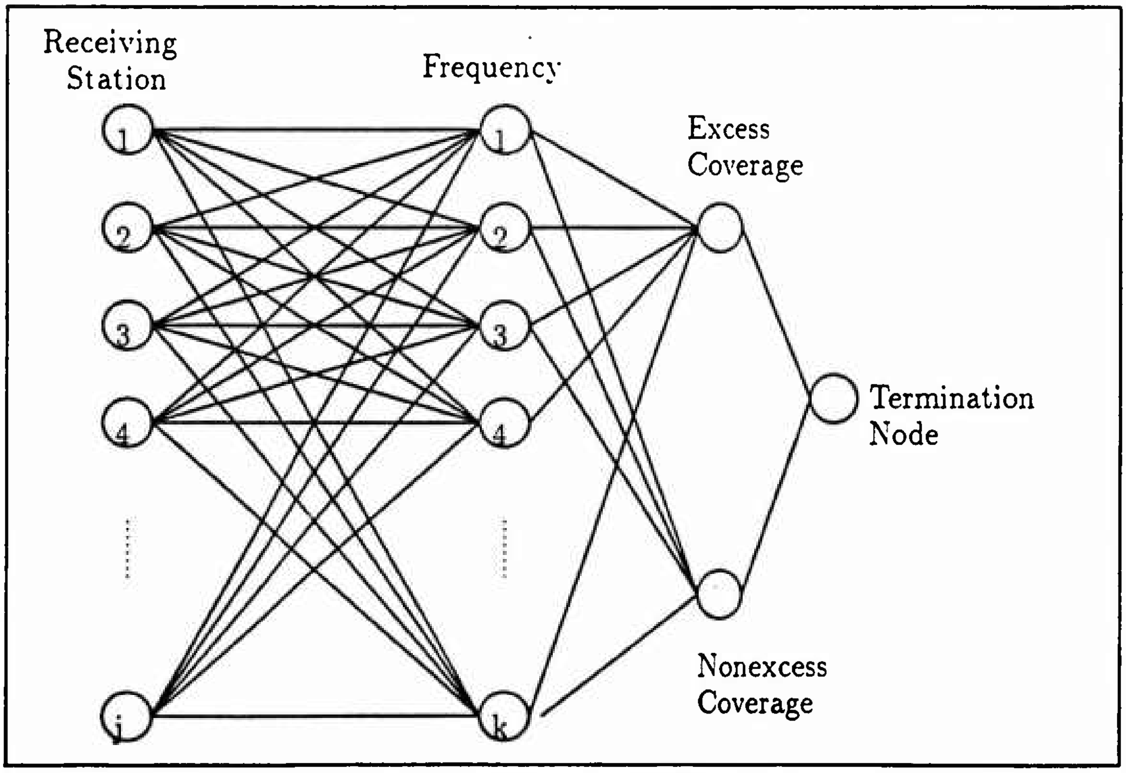
$$\lambda_1 \left[ \sum_k F_{ik} S_{ijk}^{DF} \right] + \lambda_2 \left[ \sum_k F_{ik} \right]$$

The requirement that each frequency be covered by at least two HFDF receivers is modelled by setting the "demand" at each of the frequencies equal to two. "Excess coverage" of the frequencies is penalized by adding two nodes to the basic network. Arcs from each of the frequencies to the first added node are capacitated with maximum flow equal to $\left\lceil \frac{\sum_j m_j}{\sum_k 1} \right\rceil - 2$ (two units of flow are removed form the network at the nodes corresponding to the frequencies). The cost associated with these arcs is zero. Arcs from each of the frequencies to the other added node are uncapacitated; however, the cost associated with these arcs is the weight, $\lambda_3$, assigned to the third objective function multiplied by $-1$. In mathematical terms, the cost associated with these arcs is given by $-\lambda_3$. Finally, a "terminal node" is included with arcs originating at each of the added nodes. Demand at the terminal node is equal to the total supply available at the receiving station nodes minus the total demand at the frequency nodes. This ensures that flow into the network equals flow out of the network.

Several network programming software packages, including MICROSOLVE OR (14), NETFLO (1), NETSIDE (15), and SAS Operations Research NETFLOW (18) were considered for use in this thesis. SAS NETFLOW was selected for the following reasons:

1. It is a commercial software package that is widely available.

Figure 2. Network Representation of the Search and Rescue Problem



2. It is capable of handling very large problems.

### 3.4 Multiobjective Optimization.

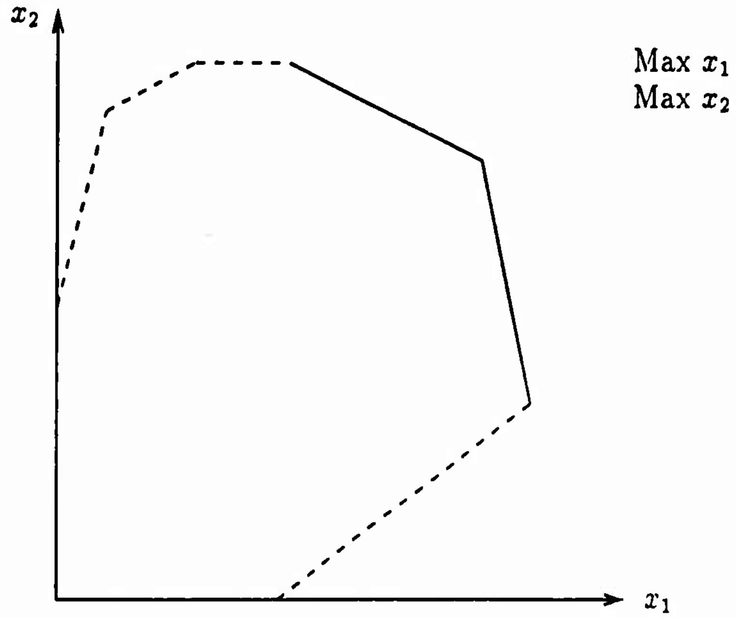
As noted in the previous section, one of the major complications in the formulation of the frequency assignment problem is the inclusion of multiple objective functions. Multiobjective decision making is traditionally applied to problems where a decision maker is confronted with conflicting objectives (20:2-3) For example, a decision maker may be faced with developing a new computer system. He may also be tasked with minimizing the cost of the new system while simultaneously maximizing the efficiency of the system. Obviously, tradeoffs are involved and, in general, the decision maker is responsible for determining what those tradeoffs should be.

In contrast, the assignment of HFDF receivers to various frequencies in the SAR network involves only one true objective – to maximize the expected number of geolocations. In this case, the multiple objectives are used as a heuristic to obtain a good frequency assignment since the true problem is intractable. Each of the objectives could be considered a heuristic to provide a frequency assignment by itself. The premise of this research, however, is that a better solution can be obtained using a multiobjective approach than could be obtained by using any of the objective functions by themselves.

The remainder of this section will describe some of the principles and solution techniques associated with multiobjective decision making.

*3.4.1 Pareto Optimality.* A point is called Pareto optimal only if there is no other point where the values of the objective functions are greater than or equal to the objective function values at that point for all of the objective functions. In other words, there are no other points that can increase the value of one objective function without decreasing the value of at least one other objective function. The set of all points that are Pareto optimal is known as the efficient frontier (23:7-16,21-32). The

Figure 3. Pareto Optimality




Pareto Optimality for a Two-dimensional Linear Feasible Region.

The efficient frontier is depicted by the solid line.

efficient frontier for a two-dimensional feasible region is shown by the solid line in figure 3.

*3.4.2 Identifying the Efficient Frontier.* The key to multiobjective optimization is identifying and evaluating the efficient frontier. Since the assignment of frequencies to HFDF receivers in the SAR network is a linear problem, two primary techniques, a multiobjective simplex method and an iterative approach, can be used to identify the efficient frontier for this problem.

*3.4.2.1 Multiobjective Simplex Method.* A variation of the simplex method described earlier can be used to identify the efficient frontier for multiobjective linear programming problems. As in the regular simplex method, an optimal solution will be a feasible corner-point solution. Additionally, any point that is Pareto optimal

will be a feasible corner-point solution or a convex combination of adjacent feasible corner-point solutions. The algorithm is also similar to the regular simplex method in that the boundary of the feasible region is explored. However, where the regular simplex method follows a single path to a single optimum (or to adjacent alternate optima), the multiobjective simplex method must identify a tree of Pareto optimal points each one of which is adjacent to at least one of the other Pareto optimal points. For a complete description of multiobjective linear programming see (20:99-114).

*3.4.2.2 ADBASE.* Only one software package, ADBASE (21), capable of identifying the entire efficient frontier for a multiobjective linear programming problem was considered for use in this thesis. ADBASE is a FORTRAN program that uses a multiobjective, revised simplex method to solve "multiple objective linear programs for all efficient extreme points and all unbounded efficient edges" (21:*i*). ADBASE has been thoroughly tested with moderate-size problems. The program is normally dimensioned to handle 15 objective functions, a total of 65 rows (constraints plus objective functions), and 110 structural variables. The program can easily be redimensioned to handle 40 objectives, 300 rows, and 300 total variables (strucutral variables plus slack, surplus and artificial variables). However, ADBASE has never been used to solve problems with more than 400 structural variables (22). Thus, the use of ADBASE for the frequency assignment problem is considered risky.

***3.4.2.3 Iterative Approach*** An alternative to the multiobjective simplex method is an iterative approach (5). For this approach, weights are assigned to each of the objective functions ($\lambda_i$ will be used to denote the weight for the $i$th objective function and the weights will hereafter be referred to as "lambdas"). The weighted objective functions are collasped into a single additive objective function and the single objective problem is solved using any appropriate linear programming software package. The process is then repeated for different lambda combinations. The efficient frontier is given by the set of solutions obtained from all possible lambda

combinations. Unfortunately, the set of all possible lambda combinations is infinite. Thus, there is no guarantee that the entire efficient frontier will be identified using the iterative approach (5).

One method that can be used to determine how many and which lambda combinations to use is to determine the maximum and minimum values possible for each of the objective functions. The single objective problem is then solved with an arbitrary lambda combination. Appropriate changes to the lambdas can be determined by comparing the values of the individual objective functions obtained by solving the single objective problem to the maximum and minimum values. For example, if an objective is at its maximum value, increasing the weight associated with that objective function will not increase the value of that objective function.

While there is no guarantee of identifying the entire efficient frontier, a major benefit of using the iterative approach in the assignment of frquencies to HFDF receivers in the search and rescue network is that efficient network algorithms can be used to solve the single objective problem.

*3.4.3 Evaluating the Efficient Frontier.* In traditional multicriteria optimization problems with conflicting objectives, the decision maker typically is presented with the available options (the Pareto optimal points) and the associated weighting combinations. The decision maker then determines which point in the efficient frontier is the best, based on his judgement and knowledge of the problem. In the case of the assignment of frequencies to HFDF receivers in the SAR network there is a single true objective to be maximized. While this objective function can be calculated, the calculation requires the evaluation of more than one billion receiving station combinations for each transmitter location/frequency combination (11:3). Although evaluating the true objective function for a single frequency tasking is within reason, evaluating the true objective function for multiple frequency taskings is not practical (11:3).

The Department of Defense has developed a FORTRAN program, EVAL, to evaluate any given frequency assignment for the HFDF receivers in the search and rescue network. This program can be used to calculate the true objective function value. EVAL can also be used to approximate the true objective function by evaluating a subset of the possible receiving station combinations for each transmitting location/frequency combination. The Department of Defense reports that good results are obtained when the number of receiving stations considered for each transmitting location/frequency combination is between eight and twelve. Additionally, the increase in the approximated objective function value associated with adding another receiving station to the subset being considered rapidly diminishes with each station that is added, while the computation time required for the evaluation doubles each time a receiving station is added (9).

By comparing the approximated objective function values for a set of possible frequency taskings for the network, the best solution from that set can be determined. Also, the Department of Defense has approximated the true objective function using ten receiving stations for each transmitting location/frequency combination for 10,000 randomly generated frequency taskings. The sample mean of the approximated objective function for those taskings is 12.614801 and the standard deviation is 0.326550 (10). These statistics can be used to evaluate whether or not the best point in the efficient frontier provides a good frequency assignment for the HFDF receivers in the SAR network.

*3.4.4 Correlation Between the Objective Functions.* If there is correlation between the objective functions, apparent paradoxes may occur in the weighting vectors. For example, if the most important objective function is highly correlated with another objective, it could be given a very small lambda value as long as the other objective is sufficiently weighted – the first objective will "go along for the ride." In other words, correlation between the objective functions sometimes yields weighting vectors that are counterintuitive (20:198-199) This implies that the

weighting of an objective function for the "best" solution does not necessaily indicate the importance of that objective.

The metric shown below can be used to evaluate the degree to which the objective functions are correlated (20:198).

$$\alpha = \arccos \left[ \frac{(c_i)^T (c_j)}{||c_i|| \, ||c_j||} \right]$$

If the value of this metric is large, the degree of correlation between the objectives is small. Notice that when $i = j$, $\alpha = 0$. In other words, an objective is perfectly correlated with itself.

The correlation between the first and third objectives and the second and third objectives is zero. This is easily seen since all coefficients corresponding to the decision variables, $x_{jk}$, are zero in the third objective function, while all coefficients corresponding to the decision variables, $Y_k$, are zero in the first and second objective functions. In contrast, the first two objective functions objective functions are expected to be highly correlated since they both include $F_{ik}$ in their coefficients. Thus, caution must be exercised when making statements about the relative worth of either of these objective functions based on the lambda combination that produces the "best" solution.

### 3.5 Solution Strategy

The solution strategy is summarized below.

1. Use ADBASE to identify the efficient frontier.

   (a) Redimension ADBASE.

   (b) Use the unimodularity of the $A$-matrix to guarantee integer solutions.

   (c) Add the set of constraints, $x_{jk} \leq 1$, to ensure the $x_{jk}$ are restricted to values of zero or one.

2. Since ADBASE has never been used to solve a problem of this size, use the iterative approach to validate results obtained from ADBASE (all points identified using the iterative approach should also be identified by ADBASE). The iterative approach will be the primary solution technique if satisfactory results are not obtained using ADBASE.

   (a) Use SAS NETFLOW to solve the single objective problems in the iterative approach.

   (b) Since the primary purpose of using the iterative approach is to validate ADBASE, use a "grid" of lambda combinations. This will provide a good sample of the efficient frontier for comparison purposes and will also provide a good starting point for determining what changes in the lambda combinations might yield improvements in the objective function if the iterative approach becomes the primary approach.

3. Use EVAL to evaluate the extreme points in the efficient frontier.

4. Compare the best solution identified in the efficient frontier to the sample mean and standard deviation obtained by the Department of Defense for the 10,000 randomly generated frequency taskings.

The results obtained by applying the solution strategy described above to the multiobjective integer linear program formulation of the frequeny assignment problem are presented and discussed in the following chapter.

# *IV. Results*

This chapter presents and discusses the results obtained by applying the solution strategy presented in Chapter III to the multiobjective integer linear programming formulation of the frequency assignment problem described in Chapter II.

The Department of Defense provided all necessary transmission and reception probabilities for twelve different two-hour time periods, however, frequency taskings were randomly generated and evaluated only for the first time period. Therefore, results for comparison and validation were only available for the first time period. The first time period, however, is considered to be representative of all of the time periods since the structure of the problem is the same for all time periods (only the probabilities change) (8). Thus, a heuristic that provides a good solution for the first time period should provide a good solution for all the time periods. Therefore, this thesis effort deals only with the first time period.

## *4.1 ADBASE*

An attempt was made to use ADBASE to solve the problem formulated in Chapter II of this document. Since documentation for ADBASE is limited with the emphasis placed on how to run the program rather than how the program works, the program had to be carefully analyzed to determine which arrays required redimensioning. Unfortunately the execution terminated prematurely. Prior to termination, eight efficient extreme points were identified; however, over 50 hours of CPU time were used on a VAX 11/785. While additional work could result in sucessfully using ADBASE to identify all of the efficient extreme points for this problem, consideration must be given to whether or not the large amount of CPU time required will result in a substantial improvement in frequency assignments for the HFDF receivers in comparison to other available techniques.

Figure 4. Results from Iterative Approach

| Combination | $\lambda_1$ | $\lambda_2$ | $\lambda_3$ | EVAL |
|---|---|---|---|---|
| 1 | 1.00 | 0.00 | 0.00 | 17.09937167 |
| 2 | 0.75 | 0.25 | 0.00 | 16.91597985 |
| 3 | 0.50 | 0.50 | 0.00 | 16.68023247 |
| 4 | 0.25 | 0.75 | 0.00 | 16.22848160 |
| 5 | 0.00 | 1.00 | 0.00 | 14.50816156 |
| 6 | 0.00 | 0.75 | 0.25 | 14.35257909 |
| 7 | 0.00 | 0.50 | 0.50 | 13.45023528 |
| 8 | 0.00 | 0.25 | 0.75 | 12.50122516 |
| 9 | 0.00 | 0.00 | 1.00 | 14.03543814 |
| 10 | 0.25 | 0.00 | 0.75 | 16.04904159 |
| 11 | 0.50 | 0.00 | 0.50 | 15.96495248 |
| 12 | 0.75 | 0.00 | 0.25 | 16.90881209 |
| 13 | 0.50 | 0.25 | 0.25 | 17.00304575 |
| 14 | 0.25 | 0.50 | 0.25 | 16.78903863 |
| 15 | 0.25 | 0.25 | 0.50 | 16.59573706 |

### 4.2 *Iterative Approach*

While ADBASE was executing, a "grid" of lambda combinations was developed to provide a good sample of the efficient frontier. The objective functions were appropriately weighted and collapsed into a single additive objective function. Each of these single objective linear programs was solved using SAS NETFLOW, taking advantage of the network structure of the problem. The CPU time required for the solution of each of these problems was approximately two minutes. If ADBASE had sucessfully executed, these solutions would have been compared to the efficient extreme points identified using ADBASE. If ADBASE had identified all of the solutions obtained using SAS NETFLOW and the iterative approach, ADBASE would have been considered to be reliable.

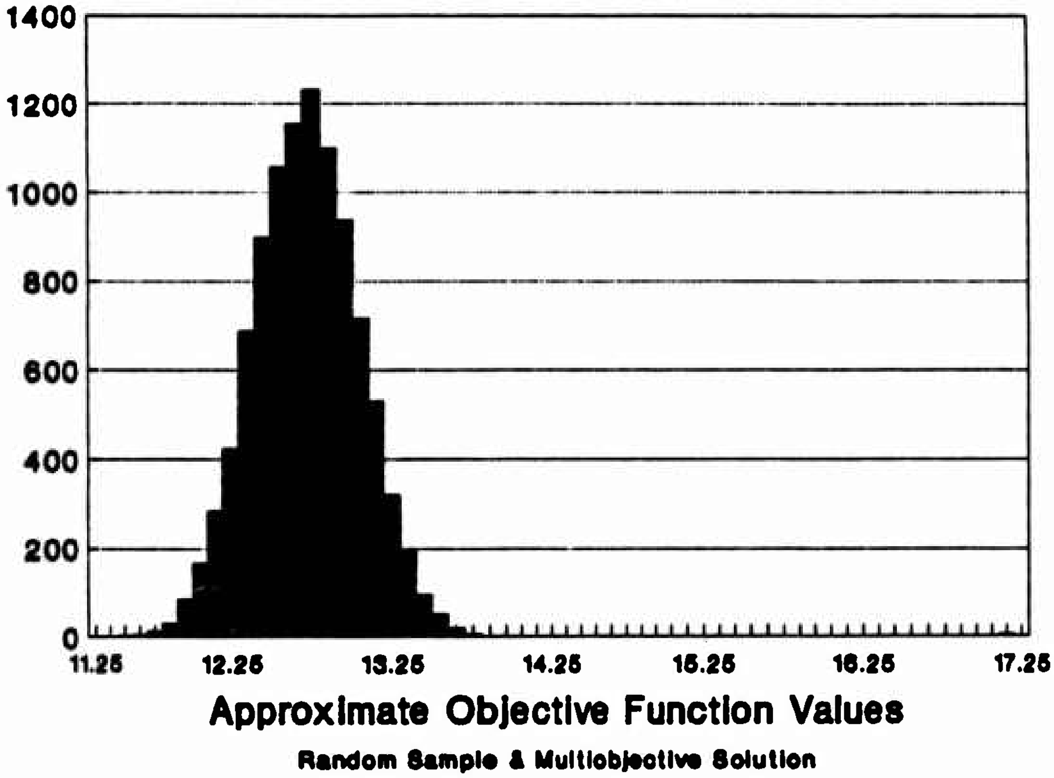
The solution from each of the single-objective network flow problems was then evaluated using the FORTRAN code, referred to here as EVAL, provided by the Department of Defense. The results are shown in figure 4.

The best solution identified in the iterative approach was produced by setting $\lambda_1$ to 1, $\lambda_2$ to 0, and $\lambda_3$ to 0 resulting in a value of 17.09937167 for EVAL. This value was compared to the results obtained by the Department of Defense for 10,000 randomly generated frequency taskings. The sample mean and the standard deviation are considered to be representative of the set of all possible frequency taskings since the size of the sample is very large (8). The sample mean value of EVAL for the 10,000 randomly generated frequency assignments is 12.614801 and the standard deviation is 0.326550. Thus, the best solution obtained from the iterative approach yields a value for EVAL that is more than 13 standard deviations better than the sample mean (see figure 5). The probability of obtaining a solution with such a large value for EVAL from a random sample is virtually zero (8). This indicates the hypothesis, that the multiobjective integer linear programming approach provides good results for the frequency assignment problem, is true.

While beyond the scope of this thesis, further exploration of the efficient frontier may result in the identification of solutions that are better than the best solution identified above. For example, the solution obtained by setting $\lambda_1$ to 0.50, $\lambda_2$ to 0.25, and $\lambda_3$ to 0.25 was nearly as good as the best solution, exceeding the sample mean by more than 13 standard deviations. Therefore, the region about this lambda combination is a good candidate for further exploration.

The conclusions reached based on the results presented in this chapter, as well as recommendations for further research, are presented in the following chapter.

Figure 5. Comparison of Solution to a Random Sample



# *V. Conclusions and Recommendations*

## *5.1 Conclusions*

A multiobjective approach can be used as a heuristic to obtain good solutions for problems with single objectives that are too complex to be solved using traditional techniques. This was demonstrated by applying a multiobjective linear integer programming approach to the problem of assigning frequencies to HFDF receivers in a search and rescue network. This approach resulted in a frequency assignment that was more than 13 standard deviations better than the results that could be expected by randomly assigning frequencies to the HFDF receivers. The probability of this occurring by chance is virtually zero.

Identifying the efficient frontier proved to be extremely challenging. While using ADBASE to identify all of the efficient extreme points appears to be feasible, the amount of CPU time required is excessive. An iterative approach, on the other hand, can take advantage of the network structure of the problem and be tailored to "home in" on the best lambda combination for any particular set of transmission and reception probabilities.

## *5.2 Recommendations*

Five major areas are recommended for further research.

1. Use the results obtained from the "grid" of lambda combinations to "home in" on a better solution or to verify that the best result identified in the grid is actually the best available solution.

2. Use the iterative approach to evaluating the efficient frontier for the remaining eleven time blocks. The results obtained should give a better indication of whether or not the second and third objective functions contribute positively to the overall objective.

3. Try different objective functions, either in combination with the three objective functions used in this thesis or by replacing one or more of them. For example, maximizing the expected number of lines of bearing provided by either HFDF or RS receivers may provide improved results. In mathematical terms, this objective function is given by

$$\sum_{i}\sum_{j}\sum_{k} F_{ik} S_{ijk}^{RS/DF}$$

4. Explore the character of the true objective function. Determine if it is convex, concave, or neither.

   If the function is convex or concave, a global maximum exists. Furthermore, if a local maximum can be identified, it will be the global maximum. This would simplify the exploration of the efficient frontier in the iterative approach since a "hill-climbing" approach could be taken to home in on the global maximum.

   If the function is neither convex nor concave, a global maximum may not exist. Furthermore, the exploration of the efficient frontier is complicated since the many local maxima may be identified without identifying a global maximum.

5. Use the results from all twelve time periods to determine if there is a lambda combination that can be used to obtain good results for all time periods. If such a lambda combination exists, it may be used to reduce the problem to a single objective network programming problem, eliminating the requirement of identifying the efficient frontier.

# Appendix A. *Frequency Taskings*

The Pareto optimal frequency assignments identified by using the iterative approach with SAS Netflow and a grid of lambdas follow.

**** TASKING 1 ****

$\lambda_1 = 1.00$ $\lambda_2 = 0.00$ $\lambda_3 = 0.00$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R15 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 2 ****

$\lambda_1 = 0.75$ $\lambda_2 = 0.25$ $\lambda_3 = 0.00$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R15 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 3 ****

$\lambda_1 = 0.50$ $\lambda_2 = 0.50$ $\lambda_3 = 0.00$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R15 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 4 ****

$\lambda_1 = 0.25 \qquad \lambda_2 = 0.75 \qquad \lambda_3 = 0.00$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R15 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 5 ****

$\lambda_1 = 0.00$ $\lambda_2 = 1.00$ $\lambda_3 = 0.00$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R02 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R03 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R15 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R28 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 6 ****

$\lambda_1 = 0.00$ $\lambda_2 = 0.75$ $\lambda_3 = 0.25$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R02 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R03 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R15 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R28 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 7 ****

$\lambda_1 = 0.00 \qquad \lambda_2 = 0.50 \qquad \lambda_3 = 0.50$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R02 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R03 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | [illegible] | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R04 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | [illegible] | [illegible] | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | [illegible] | [illegible] | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R15 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R17 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R19 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R20 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R28 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

## **** TASKING 8 ****

$\lambda_1 = 0.00 \qquad \lambda_2 = 0.25 \qquad \lambda_3 = 0.75$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R02 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R03 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R04 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| R15 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| R17 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| R19 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| R20 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R21 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R22 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | c | 1 | 0 | 0 | 0 | 0 |
| R23 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R28 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 9 ****

$\lambda_1 = 0.00$ $\lambda_2 = 0.00$ $\lambda_3 = 1.00$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 |
| SITE | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 |
| R01 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| R09 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| R10 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R11 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R15 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| R29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |

**** TASKING 10 ****

$\lambda_1 = 0.25$ $\lambda_2 = 0.00$ $\lambda_3 = 0.75$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| R15 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| R16 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 11 ****

$\lambda_1 = 0.50$ $\lambda_2 = 0.00$ $\lambda_3 = 0.50$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| R15 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| R16 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 12 ****

$\lambda_1 = 0.75$ $\lambda_2 = 0.00$ $\lambda_3 = 0.25$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R15 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R16 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 13 ****

$\lambda_1 = 0.50$ $\lambda_2 = 0.25$ $\lambda_3 = 0.25$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R15 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 14 ****

$\lambda_1 = 0.25$ $\lambda_2 = 0.50$ $\lambda_3 = 0.25$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R15 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R16 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**** TASKING 15 ****

$\lambda_1 = 0.25$ $\lambda_2 = 0.25$ $\lambda_3 = 0.50$

| | FREQUENCY IN MHZ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITE | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| R01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R06 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R07 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R09 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R10 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R14 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R15 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R16 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R17 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R18 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R19 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R20 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R21 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R22 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R25 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| R27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| R28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R29 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

## Appendix B. *Data Sorting*

Data for twelve time periods was provided by the Department of Defense, however, this thesis dealt only with the first time period. The FORTRAN program that follows sorts the data and writes the data from each time period to separate files.

```
      PROGRAM DATA SPLIT

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                              C
C THIS PROGRAM READS TRANSMISSION PROBABILITIES FROM A FILE     C
C CALLED TGTPTX.DAT AND PROPAGATION PROBABILITIES FROM FILES    C
C CALLED RSPAQ.DAT AND HFDFPAQD.DAT.  THE PROBABILITIES ARE     C
C THEN WRITTEN TO SEPARATE FILES ACCORDING TO TIME PERIOD.      C
C                                                              C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      DIMENSION RP(40,30,31), DP(40,30,31), F(40,31)

      OPEN (10,FILE='RSPAQ.DAT',STATUS='OLD')
      OPEN (11,FILE='HFDFPAQD.DAT',STATUS='OLD')
      OPEN (12,FILE='TGTPTX.DAT',STATUS='OLD')

      OPEN (20,FILE='F1.DAT',STATUS='NEW')
      OPEN (21,FILE='R1.DAT',STATUS='NEW')
      OPEN (22,FILE='D1.DAT',STATUS='NEW')

      OPEN (25,FILE='F2.DAT',STATUS='NEW')
      OPEN (26,FILE='R2.DAT',STATUS='NEW')
      OPEN (27,FILE='D2.DAT',STATUS='NEW')

      OPEN (30,FILE='F3.DAT',STATUS='NEW')
      OPEN (31,FILE='R3.DAT',STATUS='NEW')
      OPEN (32,FILE='D3.DAT',STATUS='NEW')

      OPEN (35,FILE='F4.DAT',STATUS='NEW')
      OPEN (36,FILE='R4.DAT',STATUS='NEW')
      OPEN (37,FILE='D4.DAT',STATUS='NEW')

      OPEN (40,FILE='F5.DAT',STATUS='NEW')
      OPEN (41,FILE='R5.DAT',STATUS='NEW')
      OPEN (42,FILE='D5.DAT',STATUS='NEW')

      OPEN (45,FILE='F6.DAT',STATUS='NEW')
      OPEN (46,FILE='R6.DAT',STATUS='NEW')
      OPEN (47,FILE='D6.DAT',STATUS='NEW')

      OPEN (50,FILE='F7.DAT',STATUS='NEW')
      OPEN (51,FILE='R7.DAT',STATUS='NEW')
      OPEN (52,FILE='D7.DAT',STATUS='NEW')

      OPEN (55,FILE='F8.DAT',STATUS='NEW')
      OPEN (56,FILE='R8.DAT',STATUS='NEW')
      OPEN (57,FILE='D8.DAT',STATUS='NEW')

      OPEN (60,FILE='F9.DAT',STATUS='NEW')
      OPEN (61,FILE='R9.DAT',STATUS='NEW')
      OPEN (62,FILE='D9.DAT',STATUS='NEW')
```

```
      OPEN (65,FILE='F10.DAT',STATUS='NEW')
      OPEN (66,FILE='R10.DAT',STATUS='NEW')
      OPEN (67,FILE='D10.DAT',STATUS='NEW')

      OPEN (70,FILE='F11.DAT',STATUS='NEW')
      OPEN (71,FILE='R11.DAT',STATUS='NEW')
      OPEN (72,FILE='D11.DAT',STATUS='NEW')

      OPEN (75,FILE='F12.DAT',STATUS='NEW')
      OPEN (76,FILE='R12.DAT',STATUS='NEW')
      OPEN (77,FILE='D12.DAT',STATUS='NEW')

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                              C
C  THIS SECTION READS PROPAGATION PROBABILITIES FOR THE RS     C
C  RECEIVERS AND WRITES THEM TO FILES ACCORDING TO TIME PERIOD C
C                                                              C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      DO 100 I=1,40
         DO 80 J=1,30
            READ (10,'(A5)') STRIP1
            READ (10,'(A5)') STRIP2
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (21,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (26,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (31,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (36,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (41,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (46,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (51,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (56,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (61,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (66,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (71,910) (RP(I,J,K), K=1,31)
            READ (10,900) (RP(I,J,K), K=1,31)
            WRITE (76,910) (RP(I,J,K), K=1,31)
   80    CONTINUE
  100 CONTINUE
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                  C
C  THIS SECTION READS PROPAGATION PROBAILITIES FOR THE HFDF        C
C  RECEIVERS AND WRITES THEM TO FILES ACCORDING TO TIME PERIOD C
C                                                                  C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      DO 200 I=1,40
         DO 180 J=1,30
            READ (11,'(A5)') STRIP1
            READ (11,'(A5)') STRIP2
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (22,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (27,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (32,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (37,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (42,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (47,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (52,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (57,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (62,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (67,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (72,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (77,910) (DP(I,J,K), K=1,31)
 180     CONTINUE
 200  CONTINUE

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                  C
C  THIS SECTION READS TRANSMISSION PROBAILITIES AND WRITES         C
C  THEM TO FILES ACCORDING TO TIME PERIOD.                          C
C                                                                  C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC


      DO 300 I=1,40
        READ (12,'(A5)') STRIP1
        READ (12,'(A5)') STRIP2
        READ (12,900) (F(I,K), K=1,31)
        WRITE (20,910) (F(I,K), K=1,31)
```

```
      READ (12,900) (F(I,K), K=1,31)
      WRITE (25,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (30,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (35,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (40,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (45,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (50,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (55,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (60,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (65,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (70,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (75,910) (F(I,K), K=1,31)
300 CONTINUE

900 FORMAT (1X,31(F3.2,2X))
910 FORMAT (1X,31(F3.2,1X))

    END
```

# Appendix C. *Input for SAS NETFLOW*

SAS NETFLOW requires a formatted input file. The FORTRAN program that follows computes the coefficients for the objective function and writes the necessary input file or SAS NETFLOW.

```
      PROGRAM DATA

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C   THIS PROGRAM COMPUTES COEFFICIENTS FOR THE OBJECTIVE       C
C   FUNCTION AND FORMATS THE PROBLEM FOR SAS INPUT.  THE        C
C   PROGRAM IS SET UP FOR A PROBLEM WITH 30 RECEIVING          C
C   STATIONS, 40 TRANSMITTING LOCATIONS, AND 31 FREQUENCIES.   C
C   THE DATA IS READ FROM A FILE CALLED R.DAT CONTAINING       C
C   PROBABILITIES OF DETECTION BY RS RECEIVERS, A FILE         C
C   CALLED D.DAT CONTAINING PROBABILITIES OF DETECTION BY      C
C   HFDF RECEIVERS, A FILE CALLED F.DAT CONTAINING             C
C   PROBABILITES OF TRANSMSSIONS FROM THE VARIOUS              C
C   TRANSMITTING LOCATIONS ON THE VARIOUS FREQUENCIES AND      C
C   A FILE CALLED M.DAT LISTING THE NUMBER OF HFDF RECEIVERS   C
C   AT EACH RECEIVING STATION.                                 C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      REAL X,Y,Z,F(40,31),RP(40,30,31),DP(40,30,31),S(40,30,31),
     &     C1(30,31),C2(30,31),CS(30,31)

      INTEGER CAP,COV,M(30),MTOTAL,SUP(30,31),DEM(30,31),C(30,31),D,
     &        ECAP,DEMAND


      OPEN (9,FILE='SASIN.DAT',STATUS='OLD')
      OPEN (10,FILE='R1.DAT',STATUS='OLD')
      OPEN (11,FILE='D1.DAT',STATUS='OLD')
      OPEN (12,FILE='F1.DAT',STATUS='OLD')
      OPEN (13,FILE='M.DAT',STATUS='OLD')

      CAP = 1
      COV = 2
      ECAP = 5
      PPD = 0.999
      PPR = 0.5

      DO 100 I=1,40
         READ (12,900) (F(I,K), K=1,31)
         DO 80 J=1,30
            READ (10,900) (RP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            DO 60 K=1,31
               S(I,J,K) = PPD*DP(I,J,K) - PPR * RP(I,J,K)
   60       CONTINUE
   80    CONTINUE
  100 CONTINUE
```

```
C  COMPUTE COEFFICIENTS FOR THE FIRST TWO OBJECTIVE FUNCTIONS

      DO 200 J=1,30
         DO 180 K=1,31
            C1(J,K) = 0.0
            C2(J,K) = 0.0
            DO 160 I=1,40
               C1(J,K) = C1(J,K) + F(I,K)*S(I,J,K)
               C2(J,K) = C2(J,K) + F(I,K)
 160        CONTINUE
 180     CONTINUE
 200  CONTINUE


C  COMPUTE SUPPLY AND DEMAND FOR THE NETWORK REPRESENTATION AND
C  FORMAT FOR SAS INPUT

      MTOTAL = 0
      DO 300 J=1,30
         READ (13,*) M(J)
         MTOTAL = MTOTAL + M(J)
 300  CONTINUE
      DEMAND = MTOTAL - 31*COV

      DO 400 J=1,30
         DO 380 K=1,31
            IF (K.EQ.1) THEN
               SUP(J,K)=M(J)
            ELSE
               SUP(J,K)=0
            END IF
            IF (J.EQ.30) THEN
               DEM(J,K)=COV
            ELSE
               DEM(J,K)=0
            END IF
 380     CONTINUE
 400  CONTINUE

C  COMPUTE COEFFICIENTS FOR THE NETWORK REPRESENTATION (SAS INPUT)

      DO 1000 N=1,15
         READ (9,*) X, Y, Z

      DO 500 J=1,30
         DO 480 K=1,31
            CS(J,K) = X*C1(J,K) + Y*C2(J,K)
            C(J,K) = 10000*(CS(J,K)+.00005)
 480     CONTINUE
 500  CONTINUE
      D = Z*(-10000)
```

```
C  WRITE THE SAS INPUT FILE

      IF (N.EQ.1) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X1.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.2) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X2.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.3) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X3.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.4) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X4.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.5) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X5.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.6) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X6.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.7) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X7.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.8) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X8.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.9) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X9.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.10) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X10.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.11) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X11.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.12) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X12.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.13) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X13.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.14) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X14.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.15) OPEN (20,FILE='[KGRIFFIN.THESIS.SAS1]A1X15.SAS',
     &                   STATUS='NEW')

      WRITE (20,*) 'OPTIONS LINESIZE=78;'
      WRITE (20,*)
      WRITE (20,*) 'TITLE ''HFDF FREQUENCY ASSIGNMENTS'';'
      WRITE (20,*)
      WRITE (20,*) 'DATA XXX;'
      WRITE (20,*)
      WRITE (20,*) 'INPUT INNODE $ OUTNODE $ MIN CAP COST SUP DEM;'
      WRITE (20,*)
      WRITE (20,*) 'CARDS;'
      WRITE (20,*)

      DO 600 J=1,30
         DO 580 K=1,31
            WRITE (20,910) J, 'R', K, 'F', '.', CAP, C(J,K),
     &                     SUP(J,K), DEM(J,K)
 580     CONTINUE
 600  CONTINUE
```

```
      DO 700 K=1,31
         WRITE (20,920) K, 'F', 'E', '.', '.', D, '0', '0'
 700  CONTINUE

      DO 800 K=1,31
         WRITE (20,930) K, 'F', 'S', '.', ECAP,'0', '0', '0'
 800  CONTINUE

      WRITE (20,940) 'E', 'T', '.', '.', '0', '0', '0'
      WRITE (20,950) 'S', 'T', '.', '.', '0', '0', DEMAND
      WRITE (20,*) ';'
      WRITE (20,*)
      WRITE (20,*) 'PROC NETFLOW MAXIMUM DATA=XXX;'
      WRITE (20,*) '     HEADNODE OUTNODE;'
      WRITE (20,*) '     TAILNODE INNODE;'
      WRITE (20,*) '     MINFLOW MIN;'
      WRITE (20,*) '     CAPACITY CAP;'
      WRITE (20,*) '     COST COST;'
      WRITE (20,*) '     SUPPLY SUP;'
      WRITE (20,*) '     DEMAND DEM;'
      WRITE (20,*)
      WRITE (20,*) 'PROC PRINT;'

      CLOSE (20)

1000  CONTINUE

 900  FORMAT(1X,31(F3.2,1X))
 910  FORMAT(1X,2(I2,A1,1X),A1,1X,I2,1X,I7,1X,2(I2,1X))
 920  FORMAT(1X,I2,4(A1,1X),I6,1X,2(A1,1X))
 930  FORMAT(1X,I2,3(A1,1X),I2,1X,3(A1,1X))
 940  FORMAT(1X,7(A1,1X))
 950  FORMAT(1X,6(A1,1X),I3)

      END
```

# Appendix D. *Input for ADBASE*

ADBASE requires a formatted input file. The FORTRAN program below computes the coefficients for the objective functions and writes the necessary input file for ADBASE.

```
      PROGRAM DATA

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C  THIS PROGRAM COMPUTES COEFFICIENTS FOR THE OBJECTIVE      C
C  FUNCTION AND FORMATS THE PROBLEM FOR ADBASE INPUT.  THE   C
C  PROGRAM IS SET UP FOR A PROBLEM WITH 30 RECEIVING         C
C  STATIONS, 40 TRANSMITTING LOCATIONS, AND 30 FREQUENCIES.  C
C  THE DATA IS READ FROM A FILE CALLED R.DAT CONTAINING      C
C  PROBABILITIES OF DETECTION BY RS RECEIVERS, A FILE        C
C  CALLED D.DAT CONTAINING PROBABILITIES OF DETECTION BY     C
C  HFDF RECEIVERS, A FILE CALLED F.DAT CONTAINING            C
C  PROBABILITES OF TRANSMSSIONS FROM THE VARIOUS             C
C  TRANSMITTING LOCATIONS ON THE VARIOUS FREQUENCIES AND     C
C  A FILE CALLED M.DAT LISTING THE NUMBER OF HFDF RECEIVERS  C
C  AT EACH RECEIVING STATION.                                C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      REAL F(40,31),RP(40,30,31),DP(40,30,31),S(40,30,31),PPD,PPR,
     &        C1(30,31),C2(30,31),CS(30,31),C3,A(4),B,M(30),MJ(4),
     &        CJ(4)

      INTEGER NOMB,NOBJS,N1,IK,IE,IS,IFASE0,NGRAYS,AK,
     &        BK,AE,BE,AS,BS,CK,P(4),Q(4),ALPHA

      CHARACTER TIME

      OPEN (13,FILE='M.DAT',STATUS='OLD')

C     DO 2500 L=1,12

C     IF (L.EQ.1) THEN
          TIME='00-02'
          OPEN (10,FILE='R1.DAT',STATUS='OLD')
          OPEN (11,FILE='D1.DAT',STATUS='OLD')
          OPEN (12,FILE='F1.DAT',STATUS='OLD')
          OPEN (21,FILE='TEST.IFI',STATUS='NEW')
C     END IF

      NOBJS = 3
      NOMB = 1
      N1 = 961
      IK = 991
      IE = 0
      IS = 31
      IFASE0 = 0
      NGRAYS = 45
      PPD = 0.999
      PPR = 0.5
      AK = 2821
      BK = 991
      AE = 0
      BE = 0
```

```
      AS = 930
      BS = 31
      CK = 0

      DO 50 J=1,30
         READ (13,*) M(J)
   50 CONTINUE

      DO 100 I=1,40
         READ (12,900) (F(I,K), K=1,31)
         DO 80 J=1,30
            READ (10,900) (RP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            DO 60 K=1,31
               S(I,J,K) = PPD*DP(I,J,K) - PPR * RP(I,J,K)
   60       CONTINUE
   80    CONTINUE
  100 CONTINUE

C  COMPUTE COEFFICIENTS FOR THE FIRST TWO OBJECTIVE FUNCTIONS

      DO 200 J=1,30
         DO 180 K=1,31
            C1(J,K) = 0.0
            C2(J,K) = 0.0
            DO 160 I=1,40
               C1(J,K) = C1(J,K) + F(I,K)*S(I,J,K)
               C2(J,K) = C2(J,K) + F(I,K)
  160       CONTINUE

C  COUNT THE NONZERO COST COEFFICIENTS

            IF (C1(J,K) .NE. 0.0) CK=CK+1
            IF (C2(J,K) .NE. 0.0) CK=CK+1
  180    CONTINUE
  200 CONTINUE

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C    WRITE THE ADBASE .IFI INPUT FILE                       C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      WRITE (21,*) '  INPUT FILE FOR TIME PERIOD  ', TIME
      WRITE (21,950) NOMB,NOBJS,N1,IK,IE,IS,IFASE0,NGRAYS

C  WRITE THE COUNT CARD FOR SLACK CONSTRAINTS (LESS THAN)

      WRITE (21,960) AK
```

```
C  WRITE THE NONZERO COEFFICIENTS FOR THE SLACK CONSTRAINTS

C          X(J1)+X(J2)+...+X(J31) <= M(J)

      KOUNT=0
      DO 1000 J=1,30
         DO 880 K=1,31
            KOUNT=KOUNT+1
            P(KOUNT)=J
            Q(KOUNT)=(J-1)*31+K
            A(KOUNT)=1.0
            IF (KOUNT.EQ.4) THEN
               WRITE (21,970) (P(KOUNT),Q(KOUNT),A(KOUNT), KOUNT=1,4)
               KOUNT=0
            END IF
  880    CONTINUE
 1000 CONTINUE

C          X(1K)+X(2K)+...+X(30K)-Y(K) <= 7

      DO 1100 K=1,31
         DO 1080 J=1,30
            KOUNT=KOUNT+1
            P(KOUNT)=K+30
            Q(KOUNT) = ((J-1)*31)+K
            A(KOUNT)=1.0
            IF (KOUNT.EQ.4) THEN
               WRITE (21,970) (P(KOUNT), Q(KOUNT), A(KOUNT), KOUNT=1,4)
               KOUNT = 0
            END IF
 1080    CONTINUE
         KOUNT = KOUNT+1
         P(KOUNT)=K+30
         Q(KOUNT) = 930 + K
         A(KOUNT)= -1.0
         IF (KOUNT.EQ.4) THEN
            WRITE (21,970) (P(KOUNT), Q(KOUNT), A(KOUNT), KOUNT=1,4)
            KOUNT = 0
         END IF
 1100 CONTINUE

C          X(JK) <= 1

      DO 1200 J=1,930
         KOUNT = KOUNT+1
         P(KOUNT)=J+61
         Q(KOUNT)=J
         A(KOUNT)=1.0
         IF (KOUNT.EQ.4) THEN
            WRITE (21,970) (P(KOUNT),Q(KOUNT),A(KOUNT), KOUNT=1,4)
            KOUNT=0
         END IF
 1200 CONTINUE
```

```
      IF (KOUNT.NE.0) THEN
         K=KOUNT
         WRITE (21,970) (P(KOUNT),Q(KOUNT),A(KOUNT), KOUNT=1,K)
      END IF

C  WRITE THE COUNT CARD FOR THE RIGHT-HAND SIDE FOR SLACK CONSTRAINTS M(J)

      WRITE (21,960) BK

C  WRITE RIGHT-HAND SIDE VALUES FOR SLACK CONSTRAINTS

      KOUNT=0
      DO 1300 J=1,30
         KOUNT=KOUNT+1
         P(KOUNT)=J
         A(KOUNT)=M(J)
         IF (KOUNT.EQ.4) THEN
            WRITE(21,980) (P(KOUNT), A(KOUNT), KOUNT=1,4)
            KOUNT=0
         END IF
 1300 CONTINUE

      DO 1400 K = 1,31
         KOUNT = KOUNT+1
         P(KOUNT)=K+30
         A(KOUNT)=7.0
         IF (KOUNT.EQ.4) THEN
            WRITE (21,980) (P(KOUNT), A(KOUNT), KOUNT=1,4)
            KOUNT=0
         END IF
 1400 CONTINUE

      DO 1500 J=1,930
         KOUNT=KOUNT+1
         P(KOUNT)=J+61
         A(KOUNT)=1.0
         IF (KOUNT.EQ.4) THEN
            WRITE (21,980) (P(KOUNT),A(KOUNT), KOUNT=1,4)
            KOUNT=0
         END IF
 1500 CONTINUE

      IF (KOUNT.NE.0) THEN
         K=KOUNT
         WRITE(21,980) (P(KOUNT),A(KOUNT), KOUNT=1,K)
      END IF

C  WRITE THE COUNT CARD FOR EQUALITY CONSTRAINTS

      WRITE (21,960) AE

C  WRITE THE COUNT CARD FOR THE RHS FOR THE EQUALTIY CONSTRAINTS

      WRITE (21,960) BE
```

```
C  WRITE THE COUNT CARD FOR THE SURPLUS CONSTRAINTS

      WRITE (21,960) AS

C  WRITE THE NONZERO COEFFICIENTS FOR THE SURPLUS CONSTRAINTS

C      X(1K)+X(2K)+...+X(30K) >= 2

      B=1.0
      KOUNT=0
      DO 1600 K=1,31
         DO 1580 J=1,30
            KOUNT = KOUNT+1
            P(KOUNT)=K
            Q(KOUNT)=((J-1)*31)+K
            IF (KOUNT.EQ.4) THEN
               WRITE (21,970) (P(KOUNT), Q(KOUNT), B, KOUNT=1,4)
               KOUNT=0
            END IF
 1580    CONTINUE
 1600 CONTINUE

      IF (KOUNT.NE.0) THEN
         K=KOUNT
         WRITE (21,970) (P(KOUNT),Q(KOUNT),B, KOUNT=1,K)
      END IF

C  WRITE THE COUNT CARD FOR THE RHS FOR THE SURPLUS CONSTRAINTS

      WRITE (21,960) BS

C  WRITE RIGHT-HAND SIDE VALUES FOR THE SURPLUS CONSTRAINTS

      B=2.0
      KOUNT = 0
      DO 1700 K=1,31
         KOUNT=KOUNT+1
         P(KOUNT)=K
         IF (KOUNT.EQ.4) THEN
            WRITE (21,980) (P(KOUNT), B, KOUNT=1,4)
            KOUNT=0
         END IF
 1700 CONTINUE

      IF (KOUNT.NE.0) THEN
         K=KOUNT
         WRITE (21,980) (P(KOUNT),B, KOUNT=1,K)
      END IF

C  WRITE THE COUNT CARD FOR NONZERO COST COEFFICIENTS

      CK = CK + 31
      WRITE (21,960) CK
```

```
C  WRITE THE NONZERO COST COEFFICIENTS

C      OBJECTIVE FUNCTION ONE

      KOUNT=0
      DO 1800 J=1,30
         DO 1780 K=1,31
            IF (C1(J,K) .NE. 0.0) THEN
               KOUNT=KOUNT+1
               P(KOUNT)=1
               Q(KOUNT)=(J-1)*31+K
               CJ(KOUNT)=C1(J,K)
               IF (KOUNT.EQ.4) THEN
                  WRITE(21,970) (P(KOUNT), Q(KOUNT),
     &                              CJ(KOUNT), KOUNT=1,4)
                  KOUNT = 0
               END IF
            END IF
 1780    CONTINUE
 1800 CONTINUE

C      OBJECTIVE FUNCTION TWO

      DO 1900 J=1,30
         DO 1880 K=1,31
            IF (C2(J,K) .NE. 0.0) THEN
               KOUNT=KOUNT+1
               P(KOUNT)=2
               Q(KOUNT)=(J-1)*31+K
               CJ(KOUNT)=C2(J,K)
               IF (KOUNT.EQ.4) THEN
                  WRITE(21,970) (P(KOUNT), Q(KOUNT),
     &                              CJ(KOUNT), KOUNT=1,4)
                  KOUNT=0
               END IF
            END IF
 1880    CONTINUE
 1900 CONTINUE

C       OBJECTIVE FUNCTION THREE

      DO 2000 K=1,31
         KOUNT=KOUNT+1
         P(KOUNT)=3
         Q(KOUNT)=930+K
         CJ(KOUNT)= -1.0
         IF (KOUNT.EQ.4) THEN
            WRITE (21,970) (P(KOUNT), Q(KOUNT),
     &                         CJ(KOUNT), KOUNT=1,4)
            KOUNT=0
         END IF
 2000 CONTINUE
```

```
      IF (KOUNT.NE.0) THEN
        K=KOUNT
        WRITE (21,970) (P(KOUNT), Q(KOUNT),
     &                  CJ(KOUNT), KOUNT=1,K)
      END IF

C  WRITE THE COUNT CARD FOR THE ALPHA-CONSTANTS

      ALPHA = 0
      DO 2100 K=1,3
         WRITE (21,960) ALPHA
 2100 CONTINUE

      CLOSE (10)
      CLOSE (11)
      CLOSE (12)
      CLOSE (21)

C 2500 CONTINUE

  900 FORMAT(1X,31(F3.2,1X))
  950 FORMAT(1X,8I8)
  960 FORMAT(1X,I8)
  970 FORMAT(1X,4(2I4,F12.8))
  980 FORMAT(1X,4(I4,4X,F12.8))

      END
```

## *Bibliography*


1. Ali, Agha Iqbal and Jeffrey L. Kennington. *NETFLO User's Guide*. Technical Report 88-OR-22, University of Massachusetts at Amherst and Southern Methodist University, 1988.

2. Bazarra, Mokhtar S. and John J. Jarvis. *Linear Programming and Network Flows*. New York: John Wiley and Sons, 1977.

3. Bertsimas, Dimitris J. *Probabilistic Combinatorial Optimization Problems*. Technical Report No. 194, Massachusetts Institute of Technology, 1988.

4. Brandeau, Margaret L. and Samuel S. Chiu. "An Overview of Representative Problems in Location Research," *Management Science*, *35*:645–674 (June 1989).

5. Chan, Yupo. Multicriteria Decision Making, OPER 761, Class Lecture, 18 October 1989.

6. Daskin, Mark S. "A Maximum Expected Covering Location Model: Formulation, Properties, and Heuristic Solution," *Transportation Science*, *17*:48–70 (February 1983).

7. Dearing, P. M. "Location Problems," *Operations Research Letters*, *4*:95–98 (October 1985).

8. Drake, Capt David A. Conversation at AFIT, 12 December 1989.

9. Drake, Capt David A. Telephone Conversation, 22 February 1990.

10. Drake, Capt David A. Telephone Conversation, 12 February 1990.

11. Drake, Capt David A. and Alfred B. Marsh. "High Frequency Direction Finding Problem Statement." Distributed to students and faculty in the Department of Operational Sciences. Air Force Institute of Technology (AU), Wright-Patterson AFB OH, Spring 1989.

12. Hillier, Frederick S. and Gerald J. Lieberman. *Introduction to Operations Research*. Oakland, California: Holden-Day, Inc., 1986.

13. Hogan, Kathleen and Charles Revelle. "Concepts and Applications of Backup Coverage," *Management Science*, *32*:1434–1444 (November 1986).

14. Jensen, Paul A. *MICROSOLVE Operations Research Revised Edition Student Version*. Oakland, California: Holden-Day Inc., 1985.

15. Kennington, Jeffrey L. and Alan Whisman. *NETSIDE User's Guide*. Technical Report 86-OR-01, Southern Methodist University and Navy Personnel Research and Development Center, 1988.

16. Louveaux, Francois S. and Jacques-Francois Thisse. "Production and Location on a Network Under Demand Uncertainty," *Operations Research Letters*, *4*:145–149 (December 1985).

17. Nemhauser, George L. and Laurence A. Wolsey. *Integer and Combinatorial Optimization*. New York: John Wiley and Sons, 1988.

18. SAS Institute, Inc. *SAS OR User's Guide, Version 5 Edition*. Cary, North Carolina: SAS Insitute Inc., 1985.

19. Seppala, Yrjo. "On a Stochastic Multi-Facility Location Problem," *AIIE Transactions*, 7:56–62 (March 1975).

20. Steuer, Ralph E. *Mulitple Criteria Optimization: Theory, Computation, and Application*. New York: John Wiley and Sons, 1986.

21. Steuer, Ralph E. *ADBASE Operating Manual (Version 9/89)*. Technical Report, University of Georgia, 1989.

22. Steuer, Ralph E. Telephone Conversation, 22 February 1990.

23. Yu, Po-Lung. *Multiple-Criteria Decision Making*. New York: Plenum Press, 1985.

## *Vita*

Captain Krista E. Johnson was born on 18 October 1954. She graduated from St. Cloud Technical High School, St. Cloud, Minnesota in 1972. In 1974, she enlisted in the United States Air Force. She served as a weather observer at Clark AB, Phillipines from 1974 to 1976 and at Kingsley Field, Oregon from 1976 to 1978. From 1978 to 1980, she was assigned to the "Typhoon Chasers" at Andersen AFB, Guam as a dropsonde systems operator. From 1980 to 1982, she served as a weather forecaster with assignments to Air Force Global Weather Central, Offutt AFB, Nebraska and to Detachment 11, 7 Weather Wing, Hill AFB, Utah. Captain Johnson entered Air Force ROTC at Weber State College, Ogden, Utah in 1982. She graduated with a Bachelor of Science in Mathematics and was commissioned in 1984. In 1984, Capt Johnson was assigned to the Rome Air Development Center at Griffiss AFB, New York where she served as Division Management Officer, SDI Resource Analyst, Protocol Officer, and Director, Programs Reviews. She entered the School of Engineering, Air Force Institute of Technology, in August 1988.

[redacted] PII Redacted

# REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED

1b. RESTRICTIVE MARKINGS:

2a. SECURITY CLASSIFICATION AUTHORITY:

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE:

3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited

4. PERFORMING ORGANIZATION REPORT NUMBER(S): AFIT/GOR/ENS/90M-9

5. MONITORING ORGANIZATION REPORT NUMBER(S):

6a. NAME OF PERFORMING ORGANIZATION: School of Engineering

6b. OFFICE SYMBOL (If applicable):

7a. NAME OF MONITORING ORGANIZATION:

6c. ADDRESS (City, State, and ZIP Code): Air Force Institute of Technology
Wright-Patterson AFB OH 45433-6583

7b. ADDRESS (City, State, and ZIP Code):

8a. NAME OF FUNDING/SPONSORING ORGANIZATION: Department of Defense

8b. OFFICE SYMBOL (If applicable):

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER:

8c. ADDRESS (City, State, and ZIP Code): Attn: R06
9800 Savage Road
Ft George G Meade MD 20755

10. SOURCE OF FUNDING NUMBERS

| PROGRAM ELEMENT NO. | PROJECT NO. | TASK NO. | WORK UNIT ACCESSION NO. |
|---|---|---|---|
| | | | |

11. TITLE (Include Security Classification): FREQUENCY ASSIGNMENTS FOR HFDF RECEIVERS IN A SEARCH AND RESCUE NETWORK

12. PERSONAL AUTHOR(S): Krista E. Johnson, B.S., Capt, USAF

13a. TYPE OF REPORT: MS Thesis

13b. TIME COVERED: FROM ________ TO ______

14. DATE OF REPORT (Year, Month, Day): 1990 March

15. PAGE COUNT: 85

16. SUPPLEMENTARY NOTATION:

17. COSATI CODES

| FIELD | GROUP | SUB-GROUP |
|---|---|---|
| 12 | 04 | |
| 25 | 02 | |

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number): Mathematical Programming; Multiobjective Optimization; Integer Programming; Network Flows; Search and Rescue

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A multiobjective linear integer programming approach is applied to the problem of assigning frequencies to HFDF receivers in a search and rescue network in order to maximize the expected number of geolocations of vessels in distress. An iterative approach is used to solve the multiobjective linear programming problem. The individual objective functions are weighted and combined in a single additive objective function. The resulting problem is expressed as a network programming problem and solved using SAS NETFLOW. The process is repeated with different weightings for the objective functions. The solutions obtained are evaluated using a FORTRAN program provided by the Department of Defense. The best solution provided an approximate expected number of geolocations that was more than 13 standard deviations better than the mean expected number of geolocations obtained from 10,000 random frequency assignments.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL: Yupo Chan, Professor

22b. TELEPHONE (Include Area Code): (513) 255-3362

22c. OFFICE SYMBOL: AFIT/ENS